Kepler - Bug #3150

Need to add PTII Python actors back into Kepler

02/14/2008 11:39 AM - Dan Higgins

Status:	Resolved	**Start date:**	02/14/2008
Priority:	Normal	**Due date:**	
Assignee:	Chad Berkley	**% Done:**	0%
Category:	actors	**Estimated time:**	0.00 hour
Target version:	1.1.0	**Spent time:**	0.00 hour
Bugzilla-Id:	3150		

Description

Ptolemy has 2 different actors that allow one to use Python scripts. These were removed from Kepler when all the actors were instantiated at startup because they greatlyt slowed down startup (by scanning all the jar files).

Several Kepler users have asked about using Python scripts in Kepler. Now that all actors are not instantiated at startup, the Python actors can be put back into Kepler, giving an additional scripting capability.

Dan Higgins - Feb 2008

History

#1 - 02/18/2008 04:26 PM - Christopher Brooks

The issue here is that Jython, the Java implementation of Python, writes a cache file for each jar file in the class path. This results in the following problems:

1) Slow start up time, see
http://bugzilla.ecoinformatics.org/show_bug.cgi?id=1999
This is the reason the Jython actors were pulled. Part of the problem is that Kepler was loading all the actors at start up, which was causing the Jython actor to create a cache file for each jar file in the classpath. I'm not sure if Kepler is still loading all the actors at startup or not.

2) The caching tends to fail with applets and other security restricted situations. This could prevent Jython actors from working in a distributed environment. If Kepler always invokes the Jython caching mechanism on startup, then Kepler could fail to start in a sandbox.

The Jython project has show some activity in 2007, perhaps the caching has changed, though a quick review of the Jython website does not show anything promising.

I just updated $PTII/lib/jython.jar to Jython2.2.1.

#2 - 02/18/2008 08:42 PM - Dan Higgins

I have added the python/jython actors back into my own copy of kepler for experimental testing. From the console output, the jython code is NOT caching ALL jar files on the class path when the actor is instantiated.(and the startup time is noticable but not severely long. It is going through some jars, but not all of them. I think we need to understand just what determines jython caching.

Dan Higgins- Feb 18, 2008

#3 - 03/13/2008 11:41 AM - Dan Higgins

The Python actor(s) now operate correctly in Kepler. Several examples workflows are now in the demos/Python directory.

Dan Higgins - March 2008

#4 - 03/27/2013 02:22 PM - Redmine Admin

Original Bugzilla ID was 3150